

DEFINE WINDOW

DEFINE WINDOW	<i>window-name</i>
SIZE	<div> <div>AUTO</div> <div>QUARTER</div> <div><i>operand1</i> * <i>operand2</i></div> </div>
BASE	<div> <div>CURSOR</div> <div> <div>TOP</div> <div>BOTTOM</div> </div> <div> <div>LEFT</div> <div>RIGHT</div> </div> <div><i>operand3</i> / <i>operand4</i></div> </div>
REVERSED	[[<i>CD</i> = <i>background-color</i>]]
TITLE	<i>operand5</i>
CONTROL	<div> <div>WINDOW</div> <div>SCREEN</div> </div>
FRAMED	<div> <div>ON</div> <div>[[<i>CD</i> = <i>frame-color</i>]] [<i>position-clause</i>]</div> <div>OFF</div> </div>

Operand	Possible Structure				Possible Formats																Referencing Permitted	Dynamic Definition
Operand1	C	S				N	P	I													yes	no
Operand2	C	S				N	P	I													yes	no
Operand3	C	S				N	P	I													yes	no
Operand4	C	S				N	P	I													yes	no
Operand5	C	S				A															yes	no

Related Statements: SET WINDOW | INPUT WINDOW='window-name' | SET CONTROL 'W'

Function

The DEFINE WINDOW statement is used to specify the size, position and attributes of a window.

A window is that segment of a logical page, built by a program, which is displayed on the terminal screen. There is always a window present, although you may not be aware of its existence: unless specified differently, the size of the window is identical to the physical size of your terminal screen.

A DEFINE WINDOW statement does not activate a window; this is done with a SET WINDOW statement or with the WINDOW clause of an INPUT statement.

Note:

There is always only **one** Natural window, that is, the most recent window. Any previous windows may still be visible on the screen, but are no longer active and are ignored by Natural. You may enter input only in the most recent window. If there is not enough space to enter input, the window size must be adjusted first.

Control of Full Screen

Even if a window is active, Natural maintains control of the full screen. This does not affect single-session systems like CICS or TSO; when Natural is running under a multiple-session system like Com-plete or Multi-pass, however, the complete Natural screen - that is, not only the currently active window, but also the Natural screen "underneath" it - will be displayed when a suspended Natural session is resumed; at the same time, the attributes of the full-screen fields partially overlaid by the window will not be influenced by the window.

window-name

The *window-name* identifies the window. The name may be up to 32 characters long. For a window name, the same naming conventions apply as for user-defined variables (see the Natural Reference documentation).

SIZE

With the SIZE clause, you specify the size of the window.

- **SIZE AUTO** - The size of the window is determined automatically by Natural at runtime. The size is determined by the data generated into the window as follows:
The number of window lines will be the number of INPUT lines generated (plus possibly the PF-key lines, message line, and infoline/statistics line).
The number of window columns is determined by the longest INPUT line: Natural scans, starting from the ends of the lines, for the rightmost significant byte in a line. This may cause an input-only or modifiable field (AD=A or AD=M) to be truncated; to avoid this, you either put a single-character text string after such a field or explicitly set the window size with "SIZE *operand1* * *operand2*".
Note: The title is not part of the window data. Therefore, if the window size has been determined as described above **and** the title is longer than the window, it will be truncated.
- **SIZE QUARTER** - The size of the window will be one quarter of the physical screen.
- **SIZE *operand1* * *operand2*** - The size of the window will be *n* lines by *n* columns. The number of lines is determined by *operand1*, the number of columns by *operand2*. Neither of the two operands must contain decimal digits.
If the window is FRAMED, the specified size will be inclusive of the frame.
The minimum possible window size is:
- without frame: 2 lines by 10 columns,
- with frame: 4 lines by 13 columns.
The maximum possible window size is the size of the physical screen.

If you omit the SIZE clause, SIZE AUTO applies by default.

Note:

On mainframe computers, Natural requires additional columns for so-called attribute bytes to be able to display data on the screen (on other platforms, such attribute bytes are not needed). Consequently, on mainframe computers the screen area overlaid by a window is wider, and the size of the page segment visible inside a window is smaller than on other platforms. Example: Assume a window whose size is defined as "SIZE 5 * 15" (that is, with a width of 15 columns):

- On mainframe computers, the screen area overlaid by the window is 16 columns; the size of what is visible inside the window is 14 columns without frame, and 10 columns with frame respectively.
- On other platforms, the screen area overlaid by the window is 15 columns; the size of what is visible inside the window is 15 columns without frame, and 13 columns with frame respectively.

BASE

With the BASE clause, you determine the position of the window on the physical screen.

- **BASE CURSOR** places the top left corner of the window at the current cursor position. The cursor position is the *physical* position of the cursor on the screen.
If the size of the window makes it impossible to place the window at the cursor position, Natural automatically places the window as close as possible to the desired position.
- **BASE TOP/BOTTOM LEFT/RIGHT** places the window at the top-left, bottom-left, top-right, or bottom-right corner respectively of the physical screen.
- **BASE *operand3*/*operand4*** - This places the top left corner of the window at the specified line/column of the physical screen. The line number is determined by *operand3*, the column number by *operand4*. Neither of the two operands must contain decimal digits.
If the size of the window makes it impossible to place the window at the specified position, you will get an error message.

If you omit the BASE clause, BASE CURSOR applies by default.

REVERSED

REVERSED will cause the window to be displayed in reverse video (if the screen used supports this feature; if it does not, REVERSED will be ignored).

REVERSED - CD=background-color

This will cause the window to be displayed in reverse video and the background of the window in the specified color (if the screen used supports these features; if it does not, the respective specification will be ignored).

For information on valid color codes, see the session parameter CD in the Natural Reference documentation.

TITLE operand5

With the TITLE clause, you may specify a heading for the window. The specified title (operand5) will be displayed centered in the top frame-line of the window. The title can be specified either as a text constant (in apostrophes) or as the content of a user-defined variable. If the title is longer than the window, it will be truncated. The title is only displayed if the window is FRAMED; if FRAMED OFF is specified for the window, the TITLE clause will be ignored.

Note:

If the title contains trailing blanks, these will be removed.

If the first character of the title is a blank, one blank will automatically be appended to the title.

CONTROL

With the CONTROL clause, you determine whether the PF-key lines, the message line and the statistics line are displayed in the window or on the full physical screen.

CONTROL WINDOW

CONTROL WINDOW causes the lines to be displayed inside the window.

CONTROL SCREEN

CONTROL SCREEN causes the lines to be displayed on the full physical screen outside the window.

If you omit the CONTROL clause, CONTROL WINDOW applies by default.

FRAMED

By default, i.e. if you omit the FRAMED clause, the window is framed. If you specify FRAMED OFF, the framing and everything attached to the frame (window title and position information) will be switched off.

The top and bottom frame lines are cursor-sensitive: where applicable, you can page forward, backward, left or right within the window by simply placing the cursor over the appropriate symbol (<, -, +, or >; see *position-clause* below) and then pressing ENTER. If no symbols are displayed, you can page backward and forward within the window by placing the cursor in the top frame line (for backward positioning) or bottom frame line (for forward positioning) and then pressing ENTER.

Note:

If the window size is smaller than 4 lines by 12 (or 13 on mainframe computers) columns, the frame will not be visible.

FRAMED - CD=frame-color

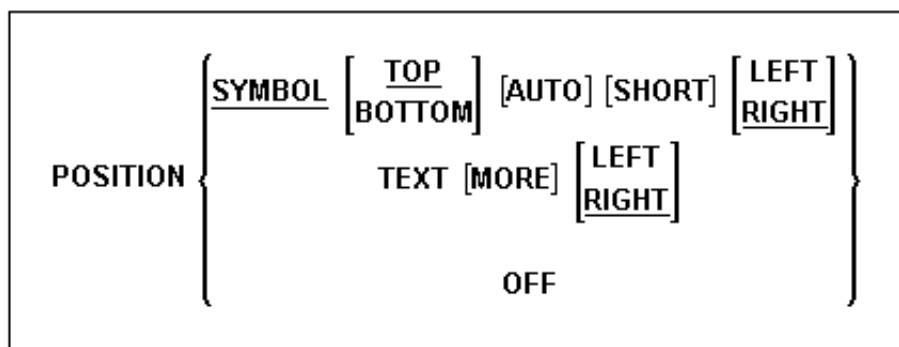
This causes the frame of the window to be displayed in the specified color (if the screen used is a color screen; if it is not, the color specification will be ignored).

For information on valid color codes, see the session parameter CD in the Natural Reference documentation.

position-clause

Note:

The POSITION clause is only evaluated on mainframe computers: on all other platforms it is ignored.



The POSITION clause causes information on the position of the window on the logical page to be displayed in the frame of the window. This applies only if the logical page is larger than the window; if it is not, the POSITION clause will be ignored. The position information indicates in which directions the logical page extends above, below, to the left and to the right of the current window.

If the POSITION clause is omitted, POSITION SYMBOL TOP RIGHT applies by default.

POSITION SYMBOL

POSITION SYMBOL causes the position information to be displayed in form of symbols: "More: < - +>". The information is displayed in the top and/or bottom frame line.

TOP/BOTTOM determines whether the position information is displayed in the top or bottom frame line.

AUTO is only applicable if the logical page is fully visible in the window as far as its horizontal size is concerned, that is, if only "-" and/or "+" are to be displayed. In this case, AUTO automatically switches from the symbols to the words "Top", "Bottom" and "More" respectively.

SHORT causes the word "More:" before the symbols "< - +>" to be suppressed.

LEFT/RIGHT determines whether the position information is displayed in the left or right part of the frame line.

POSITION TEXT

POSITION TEXT causes the position information to be displayed in text form. The information is displayed in the top and/or bottom frame line with the words "More", "Top" and "Bottom". The text is language-dependent and may also be displayed in another language if the language code is set accordingly.

POSITION TEXT MORE suppresses the words "Top" and "Bottom" and only displays the word "More" where applicable, i.e., in the top or bottom frame line or both.

LEFT/RIGHT determines whether the position information is displayed in the left or right part of the top frame line.

POSITION OFF

POSITION OFF causes the position information to be suppressed; no position information will be displayed.

Protection of Input Fields in a Window

The following rules apply to input fields (AD=A or AD=M) which are not entirely within the window:

- Input fields whose beginning is not inside the window are always made protected.
- Input fields which begin inside and end outside the window are only made protected if the values they contain cannot be displayed completely in the window. Please note that in this case it is decisive whether the *value length*, not the *field length*, exceeds the window size. Filler characters (as specified with the profile parameter FC) do not count as part of the value.

If you wish to access input fields thus protected, you have to adjust the window size accordingly so that the beginning of the field/end of the value is within the window.

Invoking Different Windows

A DEFINE WINDOW statement must not be placed within a logical condition statement block. To invoke different windows depending on a condition, use different SET WINDOW statements (or INPUT statements with a WINDOW clause respectively) in a condition.

Example

```
/* EXAMPLE 'DWDEX1': DEFINE WINDOW
DEFINE DATA LOCAL
01 #I(P3)
END-DEFINE
*
SET KEY PF1='%W<<' PF2='%W>>' PF4='%W--' PF5='%W++'
*
DEFINE WINDOW WIND1
  SIZE QUARTER
  BASE TOP RIGHT
  FRAMED ON POSITION SYMBOL AUTO
*
SET WINDOW 'WIND1'
FOR #I = 1 TO 10
  WRITE 25X #I 'THIS IS SOME LONG TEXT' #I
LOOP
*
END
```

```
> r                                     +-----More:      + >+
All      ....+....1....+....2....+....3.. ! Page      1      !
0010 /* EXAMPLE 'DWDEX1': DEFINE WIND !      !
0020 DEFINE DATA LOCAL                  !      1 THIS !
0030 01 #I(P3)                          !      2 THIS !
0040 END-DEFINE                         !      3 THIS !
0050 *                                  !      4 THIS !
0060 SET KEY PF1='%W<<' PF2='%W>>' PF !      5 THIS !
0070 *                                  !      6 THIS !
0080 DEFINE WINDOW WIND1                 !      7 THIS !
0090   SIZE QUARTER                      ! MORE      !
0100   BASE TOP RIGHT                    +-----+
0110   FRAMED ON POSITION SYMBOL AUTO
0120 *
0130 SET WINDOW 'WIND1'
0140 FOR #I = 1 TO 10
0150   WRITE 25X #I 'THIS IS SOME LONG TEXT' #I
0160 END-FOR
0170 *
0180 END
0190
0200
      ....+....1....+....2....+....3....+....4....+....5....+... S 18  L 1
```